

Techniques for Maintaining Persistent Preferences

David Oliver Skura
Michael Allen Carroll
Blair Rodericke Bishop
Jun An Yan
Siamak Safarian

Field of the Invention

[0001] The present invention relates to network transactions, and more particularly to techniques for maintaining persistent preference data for services over a network.

Background of the Invention

[0002] End users are increasingly accessing services offered by organizations over a network. More particularly, services are often browser-enabled and include interfaces that permit an end user to access the services using a World-Wide Web (WWW) browser that is interfaced to the Internet.

[0003] Many services include a variety of processing that is often time-consuming or repetitive for end-users. As a result, the services permit preferences to be stored in local locations associated with the end-users. In many cases, these preferences are stored as WWW cookies with specific names and formats that the services can locate and consume when needed. Thus, an end-user can avoid repetitive tasks at initialization with the service by saving an appropriate service preference cookie on the end-user's computing environment or device.

[004] Accordingly, when an end-user logs into a service via a WWW browser, the preference cookie is located and supplied automatically and seamlessly to the service. The service then processes the cookie to ensure that the end-user is configured in a manner that the

end-user prefers or in a service processing state that the end-user prefers.

[0005] One major drawback with cookies and locally stored preferences in general, is that if the end-user does not always access the service from the same computing environment or device, then the preference is not obtainable and not detected. Thus, the use of cookies is valuable for a specific computing environment or device, but not truly portable for the end-user. Correspondingly, when the end-user does not use the computing device that was used to create his/her initial preference cookie, the service is unable to acquire the end-user's preference during initialization and the end-user is forced to try to remember the preference or manually recreate it within the service.

[0006] Therefore, there is a need to create improved portability techniques for maintaining preferences.

Summary of the Invention

[0007] In various embodiments of this invention preferences are persistently maintained and managed. More specifically, a method for managing a preference is described. A preference is received over a network and is associated with an entity during a first transaction with a service. The preference is stored in a data store. Next, a second transaction made by the entity for the service is identified. The preference is installed on a computing device of the entity and the service is activated. The service automatically uses the preference from the computing device of the entity.

[0008] In still another embodiment of the present invention, a preference managing system is presented. The preference managing system includes a data store, a service, and a preference manager. The data store houses preferences of entities. The service is accessed by the entities, and the preference manager

records the preferences and associates the preferences with the entities. Moreover, the preference manager installs the preferences in locations that are local to the entities and expected by the service prior to the service initializing for access by the entities.

[0009] In another embodiment of the present invention, a portable preference data structure implemented in a computer-readable medium is taught. The portable preference data structure includes an entity identifier and preference data. The entity identifier and preference data combine to form a data store preference record for an entity. The record is stored externally from an environment of the entity, and the preference data is automatically transferred and installed in a local environment of the entity when the entity attempts to access a service, which consumes the preference data.

[0010] Still other aspects of the present invention will become apparent to those skilled in the art from the following description of various embodiments. As will be realized the invention is capable of other embodiments, all without departing from the present invention. Accordingly, the drawings and descriptions are illustrative in nature and not intended to be restrictive.

Brief Description of the Drawings

[0011] FIG. 1 depicts a flow diagram representing a method for managing a preference;

[0012] FIG. 2 depicts a flow diagram representing another method for managing a preference;

[0013] FIG. 3 depicts a diagram of a preference managing system;
and

[0014] FIG. 4 depicts a diagram representing a portable preference data structure.

Detailed Description of the Invention

[0015] In the following description, reference is made to the accompanying drawings that form a part hereof, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable one of ordinary skill in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that structural, logical, optical, and electrical changes may be made without departing from the scope of the present invention. The following description is, therefore, not to be taken in a limited sense, and the scope of the present invention is defined by the appended claims and their equivalents.

[0016] As used herein an entity includes an electronic representation of an end-user, an electronic representation of an electronic or computing resource (e.g., printer, server, organization, department, and the like), or an electronic application, service, or system. The entity is a requestor of a service. The service provides some functionality that the entity desires to interact during an electronic transaction. The entity accesses the service over a network. In one embodiment, the entity accesses the service indirectly over the Internet via a WWW browser application. In some embodiments, the service is integrated into a larger service or system. Thus, the service can be a stand alone and independent service or a service that is dependent upon a larger service or system.

[0017] A preference or preference data are commands, instructions, identifiers, or other data that can be processed by the service for purposes of reaching a particular processing state within the service. For example, a preference can be a database search query that is automatically processed during initialization of the service and the results automatically displayed or available to the entity. A preference need not be a single command, instruction, or identifier, such that in some embodiments a preference is a series of

commands, instructions, or identifiers that direct the service to process a number of operations and traverse through a number of service processing states in order to reach a desired end state. Thus, in some embodiments, the preference can be a history of an entity's prior transaction with the service, which is automatically processed and recreated during a new transaction on behalf of the entity.

[0018] One embodiment of the invention is implemented within the Teradata Demand Chain Management product using the Teradata data warehouse and distributed by NCR Corporation of Dayton, Ohio. Furthermore, various embodiments of this invention are presented as examples that are related to the Customer Relationship Management (CRM) and Business Analysis. These examples are illustrative and are not intended to be limiting, although the embodiments of this invention are particularly beneficial and useful in the CRM and Business Analysis industries.

[0019] FIG. 1 is a flow diagram representing one method 100 for managing a preference. The method 100 is implemented in a computer-readable medium and is accessible over a network. In one embodiment, the processing of the method 100 is implemented as a front-end interface to a service that is accessible over the network to an entity. In another embodiment, the processing of the method 100 is implemented within the processing logic of the service.

[0020] Initially, at 110, an entity engages in a transaction with a service over a network. The transaction can be viewed as a session between the entity and the service. The session can be terminated abnormally, normally, via defined events (e.g. time outs for entity inactivity with the service), and the like. During the first transaction, the entity either directly or indirectly identifies a preference that the entity desires to maintain. Direct identification can be achieved via one or more interface screens supplied within the service, where the

entity elects an option to save a preference and provides some identifier for the saved preference. Indirect identification can be achieved through the service or the processing of the method 100 detecting the actions being performed by the entity during the first transaction and recording these with a transaction identifier. In this way, indirect identification of a preference can be used to maintain an entity's processing history during the first transaction with the service.

[0021] At 120, the preference obtained during the first transaction is stored in a data store. The data store can be an electronic file, a data base, or a plurality of data bases organized as a data warehouse. In one embodiment, the preference is stored in a portable data format, such as Extensible Markup Language (XML). Of course, any intermediate or proprietary data format can be used as the native data format for the preference within the data store. The preference is recorded within the data store with an entity identifier and a preference identifier. The preference identifier can be a manual supplied description received from the entity, or it can be an automatically generated identifier, such as a first transaction identifier and the like.

[0022] At some later point in time, the entity associated with the stored preference will establish a second transaction or second session with the service. Accordingly, at 130, the processing of the method 100 identifies a second transaction between the entity and the service occurring over a network. In one embodiment, the processing of the method 100 acts a proxy of front-end interface for the service, such that the second transaction is detected during an entity's attempt to log into the service at 132 for the start of the second transaction. In other embodiments, the processing of the method 100 is invoked initially by the service immediately after the entity successfully logs into the service for the second transaction.

[0023] At 134, the previously stored preference is installed on a computing device of the entity. Installation can occur before the entity takes any action with the service during the second transaction. For example, the entity can be configured to use the saved preference as a default preference that is automatically installed as soon as the entity establishes a login to the service. In other embodiments, the preference is not installed until affirmatively instructed to do so by the entity from within the service. For example, the entity logs into the service, accesses a saved preferences menu, and manually selects a specific saved preference for installation.

[0024] In one embodiment, the entity is accessing the service over the Internet through a WWW browser. In these embodiments, the preference is installed within the entity's computing environment or computing device as a WWW cookie file at 136, such that the browser is enabled to supply the cookie to the service for processing during initialization of the service or when requested by the service.

[0025] Traditionally, cookies provide techniques for saving preferences for end-users, but cookies are limited to the computing environment in which they were created. That is, a created cookie is only useful when an end-user access a service that consumes that cookie from the computing device that initially created the cookie. Cookies are not portable from computing device to computing device, such that entities that have laptops or more than a few access computers are forced to recreate and manage cookies across multiple computers. With the teachings of this invention, this problem is solved, since an entity's preference is externally managed and populated to an entity's local computing device when needed. This permits preferences to remain persistent and become more portable than what has been conventionally achievable.

[0026] In some embodiments, the processing of the method 100 can check the local computing device of the entity to determine if the

preference is already present before installing the preference. This can more rapidly achieved with identifier's or other information associated with the preference, such that only the identifier or a check sum is checked to make a determination if the preference is already properly installed on the entity's computing device. In other embodiments, no check is made at all and if a prior preference is already installed on the entity's computing device, the processing of the method 100 simply writes over it and replaces it each time the entity establishes a new transaction or session with the service.

[0027] Again, in some embodiments, the preference is installed in a portable data format, such as XML within the entity's computing environment or on the entity's computing device. By using a portable data format, such as XML, the preference can be used from the entity's computing environment or computing device with a variety of other additional services, when translation utilities are developed to provide the preference in a native data format expected by those additional services. In this way, the preferences of this invention are even more portable and interoperable between disparate services and disparate computing environments.

[0028] Once the preference is installed on the entity's computing device or within the entity's computing environment, the service is activated at 140 where the preference is consumed or processed by the service. In one embodiment, the preference is a search query that is automatically executed by the service on behalf of the entity, such that the search results are immediately displayed to the entity for consumption. In other embodiments, the preference is a particular configuration state or processing state of the service, such that when the preference is executed the entity is placed in a particular state that it desires within the service. Thus, in some embodiments, the preference can actually be a series of operations that the service processes on behalf of the entity. For example, the preference may

be a history of a previous entity transaction or session, and when the service processes the preference the entity is placed back in a processing state of the service that the entity had obtained in that previous transaction. In still other embodiments, the preference may only affect the look and feel of the interface of presentation screens associated with the service. Thus, a preference can drive both presentations and substantive processing of the service.

[0029] It is now understood how a preference can be managed and made persistent across multiple computing environments and computing devices. The preference is externally managed and locally installed when needed. Entities can now enjoy preference portability in ways that were not previously capable with traditional techniques.

[0030] FIG. 2 is a flow diagram representing another method 200 for managing a preference. The method 200 is implemented in a computer-readable medium. The method is accessible over a network. In one embodiment, the processing of the method 200 is a stand alone application that interfaces with a service. In another embodiment, the processing of the method 200 is integrated and is an integral part of the service.

[0031] Initially, at 210, an entity logs into a service over a network. In some embodiments, the desired service is actually a sub-component of a larger service or system, such that the entity actually logs into the larger service of system and not directly into the service. Once the entity is logged into the service, either directly or indirectly, a new transaction or session is established between the entity and the service.

[0032] As soon as the entity is identified to the service via an entity identifier to some other identification mechanism (e.g., digital certificate, digital signature, and the like), processing continues at 220. At 220, the entity's identification is used to determine if the

entity is associated with any saved preference(s). Policies may also be associated with any detected saved preference(s), such that under some circumstances policies direct the processing of the method 200 to automatically obtain a preference as a default preference and execute the preference on behalf of the entity. Under other circumstances, policies can present via interface screens saved preference(s) associated with the entity and provide commands which permit the entity to manual select and process desired preference(s). Policies are configurable and can be accessible from the processing of the method 200 or alternatively from the service.

[0033] If, at 220, a saved preference(s) is detected for the entity, then, at 224, the preference(s) are retrieved from a data store. If there are no preference(s) associated with the entity, then at 222, and at any point during the transaction or session of the entity, the entity can access one or more commands or options within the service to save a preference for subsequent use by the processing of the method 200.

[0034] When a preference is obtained, the preference is processed at 230 by the service. Again, the processing of the preference can occur automatically within the service or alternatively the processing can be manually driven by entity selection as depicted at 232. Policies can be used in the manners described above to determine whether automatic or manual processing of the preference occurs.

[0035] Before the preference is processed, the preference is installed within the computing environment or on the computing device of the entity. The service actually detects and processes the preference from a defined location that is local to the entity. In one embodiment, the entity's local location is a WWW cookie that is accessible to an entity's WWW browser. Installing the preference locally to the entity, permits services that are designed to acquire preferences locally from entity's to use the teachings of the present invention. Moreover, by

installing the preferences locally during each transaction or session, the preferences become persistent and portable from disparate and different computing environments and computing devices.

[0036] At 240, the entity continues interacting with the service after the preference is processed. If the entity did not have a preference saved during a current transaction or session but saved one during the current transaction, then the next time the entity engages in a new transaction with the service that saved preference will be available for use in the manners described herein and above.

[0037] FIG. 3 is a diagram of one preference managing system 300. The preference management system 300 is implemented in and accessible from a computer-readable medium. The preference managing system 300 is accessible over a network 340. In one embodiment, the preference managing system 300 is integrated within a service 320 (not shown in FIG. 3). In an alternative embodiment, the preference managing system 300 is external to and interfaced with the service 320.

[0038] The preference managing system 300 includes a data store 310, a service 320, and a preference manager 330. The data store 310 is for housing preferences associated with entities 350. The preferences are consumed and processed by the service 320. The processing results associated with the preferences can drive processing states of the service 320 or drive the interfaces and/or presentation screens of the service 320 as it presents itself to the entities 350.

[0039] During operation of the preference managing system 300, an entity 350 accesses a service 320 over a network 340. In some embodiments, the access is a direct login to the service 320. In other embodiments, the access is a login to a different system or service to which the service 320 is a part of. For each login transaction or session with the service 320, the entity 350 can perform a number of

operations or commands associated with the service 320. A transaction or service 320 can be terminated abnormally, normally, or via defined events, such as a time out for lack of entity 350 actions within the service 320.

[0040] Upon initial interaction between the entity 35 and the service 320 during a transaction or session, the preference manager 330 is invoked. Invocation of the preference manager 330 can be automatic, such as when the preference manager 330 is acting as a proxy for some actions of the service 320. Invocation of the preference manager 330 can also be made by the service 320.

[0041] When the preference manager 330 is invoked, the preference manager obtains a preference for the entity 350 from the data store 310 and installs the preference on the entity's computing device or locally within the entity's computing environment. In one embodiment, the preference is installed as a WWW cookie file in a directory of the entity's computing device. In these embodiments, the entity 350 uses a WWW browser to access the service 320 and the cookie is accessible to the browser, such that the browser supplies the cookie when instructed to do so by the service 320.

[0042] In some embodiments, the preference manager 330 provides an identifier for the preference to the service 320, such that the service can present the preference for selection by the entity 350. When the entity selects the preference the preference selected is installed on the computing device or within the computing environment of the entity 350 and supplied to the service 320 for processing.

[0043] In an embodiment, the preference manager 330 stores the preference in the data store 310 and installs the preference local to the entity 350 in a portable data format, such as XML. This portable data format combined with schemas and translators can permit the preference to be consumed and used by other services that are

disparate from the service 320 of the preference managing system 300.

[0044] When the service 320 decides to process the preference; the preference is requested or obtained from locations that are local to the entity 350. The service can use configurable policies associated with the entities 350 and/or the preferences to make the decision as to when a preference is to be processed. In some embodiments, a default policy directs the service 320 to automatically process the preference from the locations local to the entity 350 after the entity 350 has successfully logged into the service 320. In other embodiments, the service 320 waits for the entity 350 to manually direct it to process the preference from the entity's local location.

[0045] Once the preference is processed by the service 320, the entity 350 is placed in a desired substantive processing state of the service 320 or presented with a desired presentation interface. In this way, the entity's preferences are persistent across multiple computing platforms and environments. Preferences can now be obtained from any computing device that the entity 350 decides to use when accessing the service 320. Traditionally, this has not been the case, since preferences associated with cookies were dependent upon the computing device in which they were originally created.

[0046] Moreover, if the entity 350 has no saved or previously stored preferences, then the service 320 in combination with the preference manager 330 permit the entity 350 to selectively save and store one or more desired preferences and in some instances define and associate policies with those saved preferences. Thus, an entity 350 can configure and save preferences for the service 320 in manners desired by the entity 350.

[0047] FIG. 4 is a diagram representing one portable preference data structure 400. The portable preference data structure 400 is implemented in a computer-readable medium and is accessible over

a network. The portable preference data structure 400 is stored locally to an entity 430 and externally from an entity 430 in a data store 430. The portable preference data structure 400 can be in any data format and in some embodiments is in a portable data format, such as XML.

[0048] The portable preference data structure 400 includes an entity identifier 401 and preference data 402. The entity identifier 401 and the preference data 402 combine to form a preference record that is housed for the entity 430 in a data store 420 that is externally located across a network from the entity 430. In some embodiments, the preference record also includes a service identifier, for uniquely identifying a service 410 that is associated with the preference data 402. The preference data can be commands, options, or identifiers that can be consumed or processed by the service 410 for purposes of obtaining a particular processing state of the service 410 or a particular presentation state of the service 410.

[0049] The preference data 402 is automatically transferred and installed locally to the entity 430 when the entity 430 establishes a session or transaction with the service 410. In one embodiment, when the entity 430 establishes a session with the service 410, a value for the entity identifier 401 is obtained and used to acquire an instance of the portable preference data structure 400 from the data store 420. Next, the preference data 402 is automatically installed as a cookie within a directory of the entity's computing device. The entity's browser then supplies the cookie when instructed to do so by the service 410, and the service 410 processes the preference to place the entity in a desired presentation or processing state within the service 410.

[0050] In some embodiments, the preference data 402 is an executable search query that the service 410 executes in order to obtain and present search results to the entity 430 as a default action

each time the entity 430 logs into the service 410. Moreover, since the preference data 402 is installed local to the entity 430, the entity 430 can use any computing device to access the service 410 and obtain the proper initial configuration and preference data. The service 410 need not be modified in any substantial manner, since the service 410 can still obtain preferences from the entity 430 in manners it normally expects (e.g., via WWW cookies).

[0051] Instances of the portable preference data structure 400 can be populated during interactions with between the service 410 and the entity 430. Furthermore, a single entity 430 can be associated with more than one portable preference data structure 400. In some embodiments, the portable preference data structure 400 is managed by a separate front-end interface to the service 410. In other embodiments, the portable preference data structure 400 is managed by applications integrated and integral to the service 410.

[0052] Additionally, the portable preference data structure 400 can include a series of transactions or commands within any particular instance of the preference data 402. In this way, the preference data 402 can be a history of an entity's prior session with the service 410 that was automatically or manually captured during the prior session. This history can be automatically acquired from the data store 420 and populated locally to the entity 430 and supplied to the service 410. The service 410 uses the history to put the entity 430 back in a processing state with the service 410 that the entity 430 had during a prior transaction.

[0053] The portable preference data structure 400 is managed externally and installed locally on computing devices or within computing devices associated with entities 430. In this manner, the portable preference data structure can be made persistent across multiple computing platforms and provides improved ease of use for entities 430 that access a service 410 over a network.

[0054] The above description is intended to be illustrative, and not restrictive. Many other embodiments will be apparent to those of skill in the art upon reviewing the above description. The scope of embodiments of the invention should, therefore, be determined with reference to the appended claims, along with the full scope of equivalents to which such claims are entitled.

[0055] The Abstract is provided to comply with 37 C.F.R. §1.72(b) requiring an Abstract that will allow the reader to quickly ascertain the nature and gist of the technical disclosure. It is submitted with the understanding that it will not be used to interpret or limit the scope or meaning of the claims.

[0056] In the foregoing Description of the Embodiments, various features are grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure is not to be interpreted as reflecting an intention that the claimed embodiments of the invention require more features than are expressly recited in each claim. Rather, as the following claims reflect, inventive subject matter lies in less than all features of a single disclosed embodiment. Thus the following claims are hereby incorporated into the Description of the Embodiments, with each claim standing on its own as a separate exemplary embodiment.